LAB ASSESSMENT - 2

PRASSAN

25BCE10617

Lab Assessment: Operators and Operator Precedence

Objectives

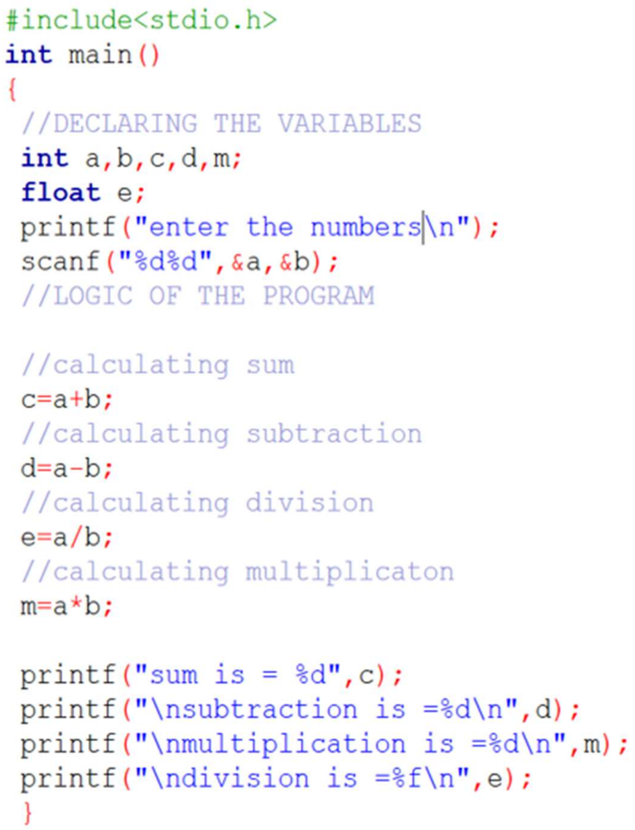
- **Understand the different types of operators in a programming language (arithmetic, relational,**
  logical, assignment, etc.).
- **Explore how operator precedence and associativity govern the evaluation of complex expressions.**
- **Practice writing, predicting, and debugging expressions containing multiple operators.**

Assessment Tasks:

Task 1: Basic Operator Usage

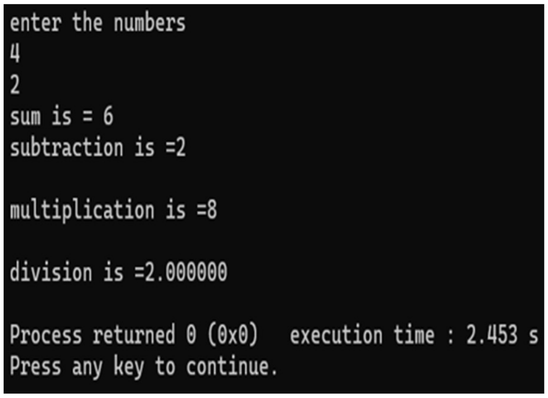
- **Write a program that accepts two integer values and demonstrates the use of basic arithmetic operators (++, −−, ∗∗, //,) by displaying the result of each operation.**
- **Output the results in a readable format (e.g., “Addition: 10 + 5 = 15”).**

SAMPLE CODE :



```
#include<stdio.h>
int main()
{
 //DECLARING THE VARIABLES
 int a,b,c,d,m;
 float e;
 printf("enter the numbers\n");
 scanf("%d%d",&a,&b);
 //LOGIC OF THE PROGRAM

 //calculating sum
 c=a+b;
 //calculating subtraction
 d=a-b;
 //calculating division
 e=a/b;
 //calculating multiplicaton
 m=a*b;

 printf("sum is = %d",c);
 printf("\nsubtraction is =%d\n",d);
 printf("\nmultiplication is =%d\n",m);
 printf("\ndivision is =%f\n",e);
 }
```

WORKING / RESULT:



```
enter the numbers
4
2
sum is = 6
subtraction is =2

multiplication is =8

division is =2.000000

Process returned 0 (0x0)   execution time : 2.453 s
Press any key to continue.
```

SAMPLE INPUT / OUTPUT :

“ADDITION : 4+2=6”

“SUBTRACTION: 4-2=2”

“MULTIPLICATION:4*2=8”

“DIVISION:4/2=2”

Task 2: Complex Expressions and Precedence

- **Ask the user to input three integer values (a, b, c).**
- **Evaluate and print the result of expressions such as:**

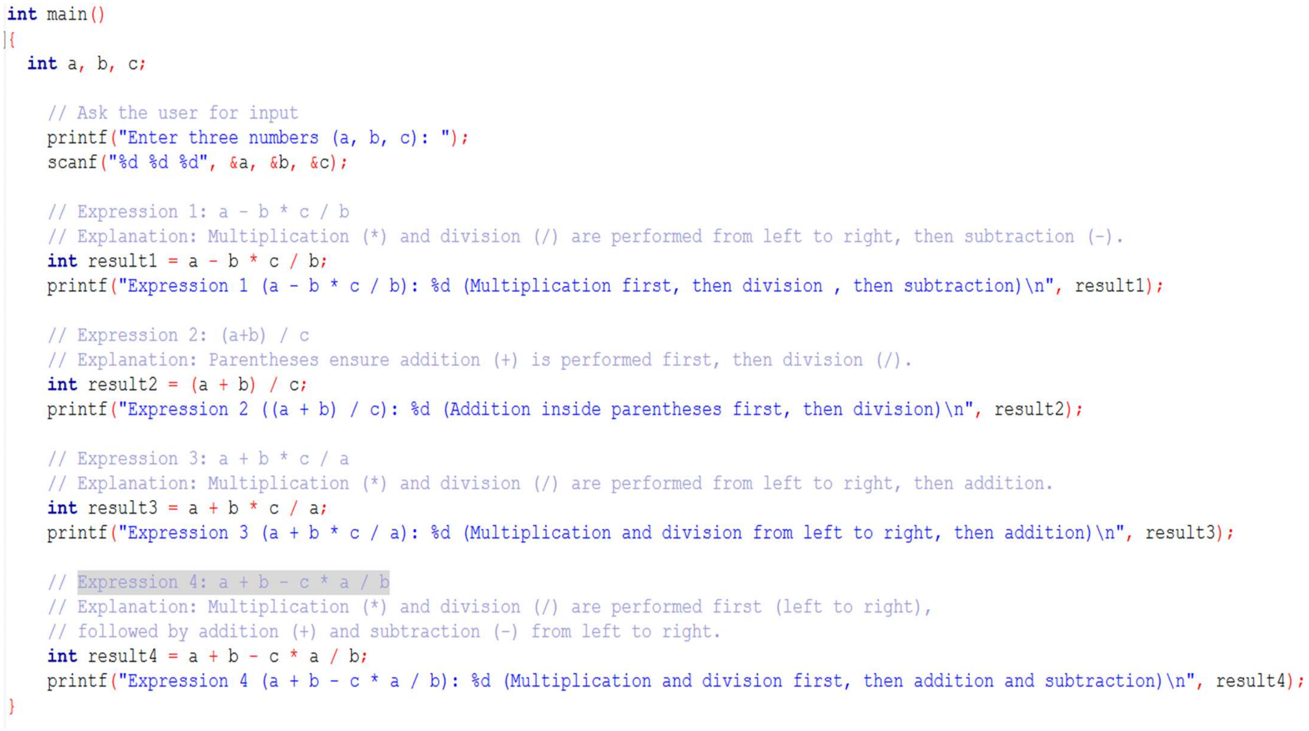
Expression 1: a - b * c / b

Expression 2: (a+b) / c

Expression 3: a + b * c / a

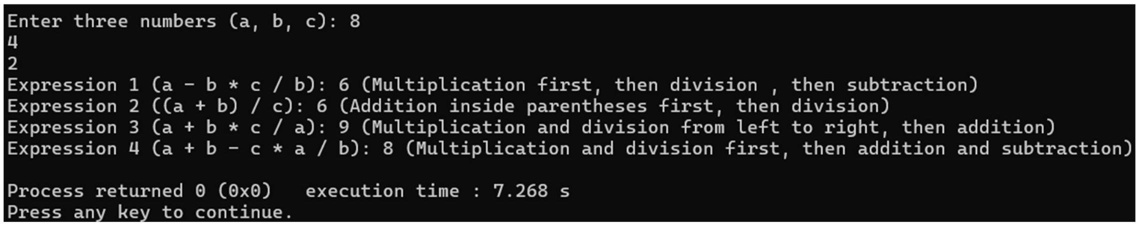
Expression 4: a + b - c * a / b

SAMPLE CODE:



```
int main()
{
  int a, b, c;

    // Ask the user for input
    printf("Enter three numbers (a, b, c): ");
    scanf("%d %d %d", &a, &b, &c);

    // Expression 1: a - b * c / b
    // Explanation: Multiplication (*) and division (/) are performed from left to right, then subtraction (-).
    int result1 = a - b * c / b;
    printf("Expression 1 (a - b * c / b): %d (Multiplication first, then division , then subtraction)\n", result1);

    // Expression 2: (a+b) / c
    // Explanation: Parentheses ensure addition (+) is performed first, then division (/).
    int result2 = (a + b) / c;
    printf("Expression 2 ((a + b) / c): %d (Addition inside parentheses first, then division)\n", result2);

    // Expression 3: a + b * c / a
    // Explanation: Multiplication (*) and division (/) are performed from left to right, then addition.
    int result3 = a + b * c / a;
    printf("Expression 3 (a + b * c / a): %d (Multiplication and division from left to right, then addition)\n", result3);

    // Expression 4: a + b - c * a / b
    // Explanation: Multiplication (*) and division (/) are performed first (left to right),
    // followed by addition (+) and subtraction (-) from left to right.
    int result4 = a + b - c * a / b;
    printf("Expression 4 (a + b - c * a / b): %d (Multiplication and division first, then addition and subtraction)\n", result4);
}
```

WORKING / RESULT:



```
Enter three numbers (a, b, c): 8
4
2
Expression 1 (a - b * c / b): 6 (Multiplication first, then division , then subtraction)
Expression 2 ((a + b) / c): 6 (Addition inside parentheses first, then division)
Expression 3 (a + b * c / a): 9 (Multiplication and division from left to right, then addition)
Expression 4 (a + b - c * a / b): 8 (Multiplication and division first, then addition and subtraction)

Process returned 0 (0x0)   execution time : 7.268 s
Press any key to continue.
```

Task 3: Predict and Verify Output

- **Provide a set of code snippets containing complex expressions with mixed operators**

Do the Following

- **Predict the output without running the code.**
- **Annotate the reason for their prediction.**
- **Run the code and compare the results, noting any differences and explaining why.**

LET US FIRST CONSIDER FOUR INTEGERS VALUE

Let say we consider: a=2, b=4, c=6, d=8

Expression 1 (easy): a + b - c + d

Explanation: Addition first then subtraction from left to right, then again addition.

Output: 6 – 6 + 8 = 0 + 8 = 8(ans)

Expression 2(medium): a + b * c – d

Explanation: Multiplication first, then addition and subtraction from left to right.

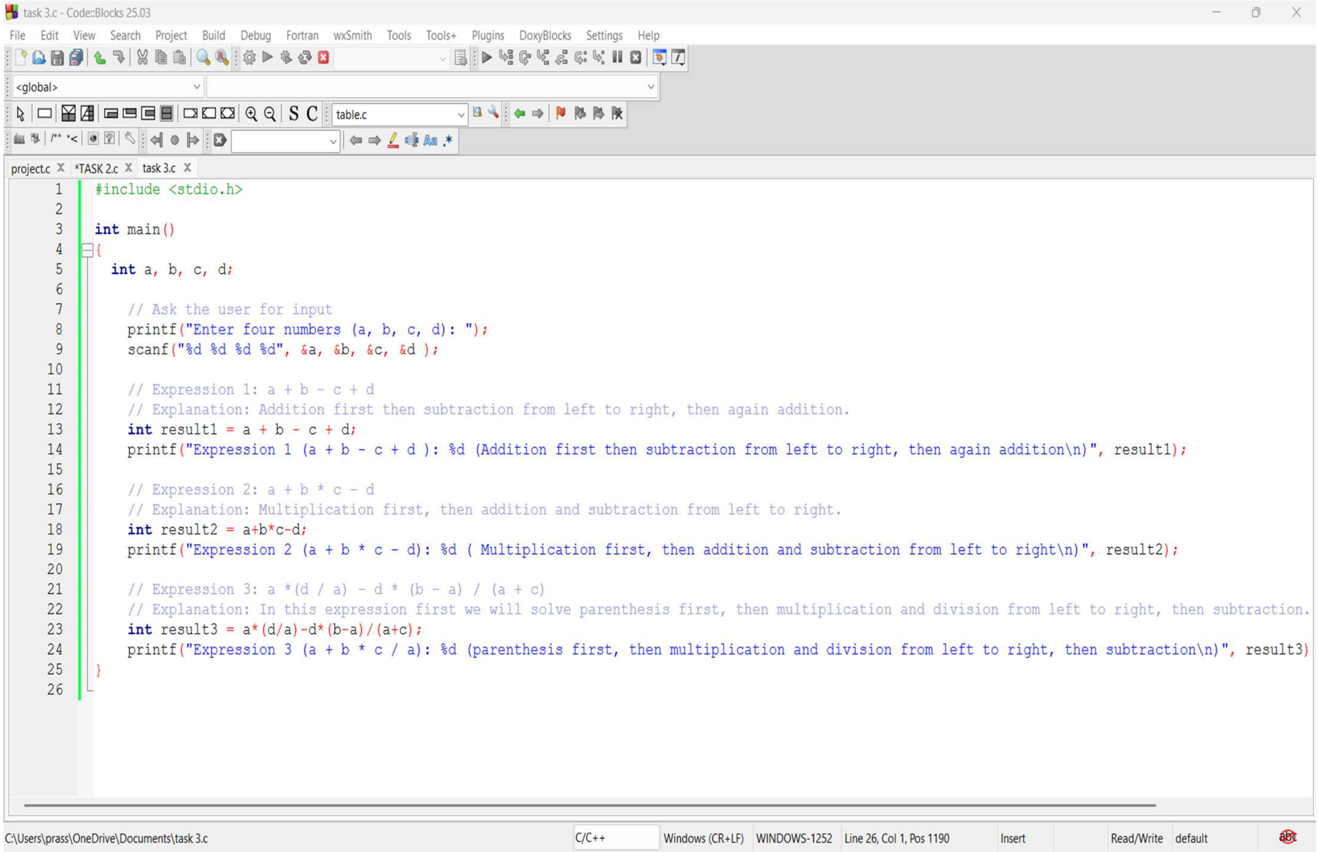
Output: 2 + 24 – 8 = 26 – 8 = 18(ans)

Expression 3(complex): a * (d / a) – d * (b – a) / (a + c)

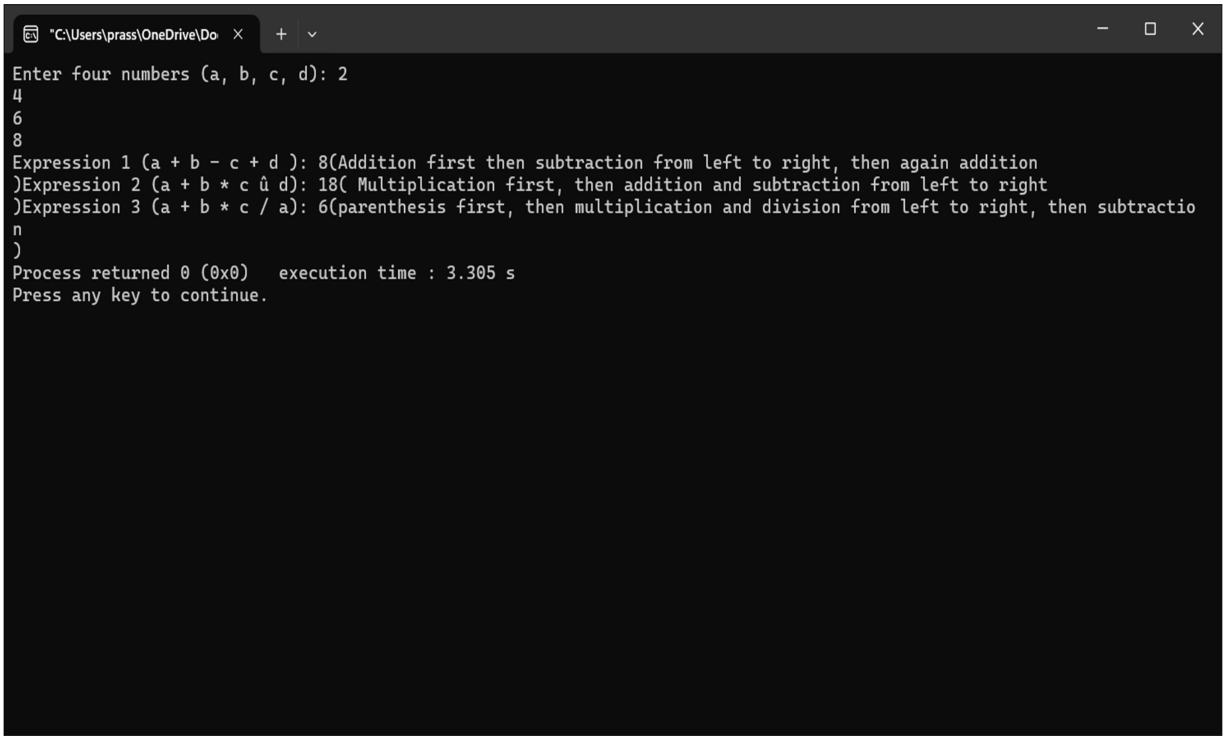
Explanation: In this expression first we will solve parenthesis, then multiplication and division from left to right, then subtraction.

Output: 2 * 4 – 8 * 2 / 8 = 8 – 16 / 8 = 8 – 2 = 6(ans)

WORKING / RESULT:


task 3.c - Code::Blocks 25.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
<global>
table.c
project.c
*TASK 2.c
task 3.c
C:\Users\prass\OneDrive\Documents\task 3.c
C/C++
Windows (CR+LF)
WINDOWS-1252
Line 26, Col 1, Pos 1190
Insert
Read/Write
default

```
#include <stdio.h>

int main()
{
  int a, b, c, d;

    // Ask the user for input
    printf("Enter four numbers (a, b, c, d): ");
    scanf("%d %d %d %d", &a, &b, &c, &d );

    // Expression 1: a + b - c + d
    // Explanation: Addition first then subtraction from left to right, then again addition.
    int result1 = a + b - c + d;
    printf("Expression 1 (a + b - c + d ): %d (Addition first then subtraction from left to right, then again addition\n)", result1);

    // Expression 2: a + b * c - d
    // Explanation: Multiplication first, then addition and subtraction from left to right.
    int result2 = a+b*c-d;
    printf("Expression 2 (a + b * c - d): %d ( Multiplication first, then addition and subtraction from left to right\n)", result2);

    // Expression 3: a *(d / a) - d * (b - a) / (a + c)
    // Explanation: In this expression first we will solve parenthesis first, then multiplication and division from left to right, then subtraction.
    int result3 = a*(d/a)-d*(b-a)/(a+c);
    printf("Expression 3 (a + b * c / a): %d (parenthesis first, then multiplication and division from left to right, then subtraction\n)", result3)
}

```



```
Enter four numbers (a, b, c, d): 2
4
6
8
Expression 1 (a + b - c + d ): 8(Addition first then subtraction from left to right, then again addition
)Expression 2 (a + b * c û d): 18( Multiplication first, then addition and subtraction from left to right
)Expression 3 (a + b * c / a): 6(parenthesis first, then multiplication and division from left to right, then subtractio
n
)
Process returned 0 (0x0)   execution time : 3.305 s
Press any key to continue.
```

Task 4: Parentheses to Control Precedence

- **Demonstrate how inserting parentheses can change the result of expressions.**
- **Let us consider three integer values a=2, b=3, c=4**

- **Expression 1: (a + b) ∗ c (a + b) ∗ c versus a + (b ∗ c) a + (b ∗ c)**

  Explanation of first half: In this expression first we will solve paranthesis (a+b), then multiplication.
  Explanation in second half: In this expression first we will solve paranthesis (b*c), then addition.
  Output in the first half: 5 * 4(5) * 4 = 5*20*4 = 400(ans)
  Output in the second half: 3 + (12)3 + 12 = 3 + 36 + 12 = 52(ans)

- **Expression 2: (a − b) / c (a − b) / c versus a − (b / c) a − (b / c)**
- Explanation of first half: In this expression first we will solve parenthesis (a+b), then multiplication and division from left to right.
- Explanation of second half: In this expression first we will solve parenthesis (b/c), then division, multiplication and subtraction.
- Output of first half: (-1) / 4 (-1) / 4 = 1
- Output of second half: 2 – (3/4)2 – 3/4 = 2 – 3/2 – 3/4 = -(1/4)ans

PRASSAN RASTOGI

25BCE10617

B11+B12+B13